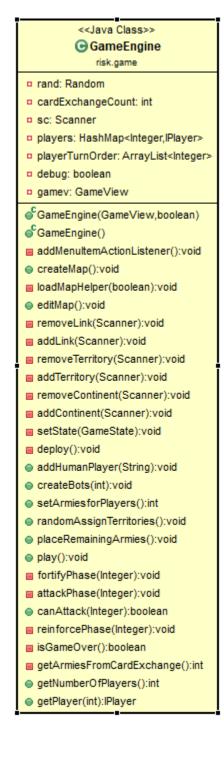


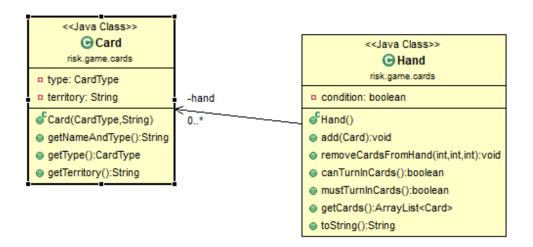
🖶 test.risk.game.cards

<<Java Package>>
test.risk.utils

<<Java Package>> ₍ test.risk.game

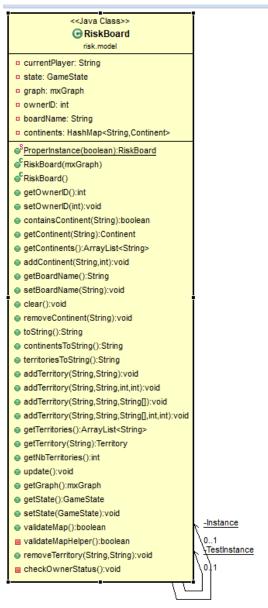
<<Java Package>> test.risk.model





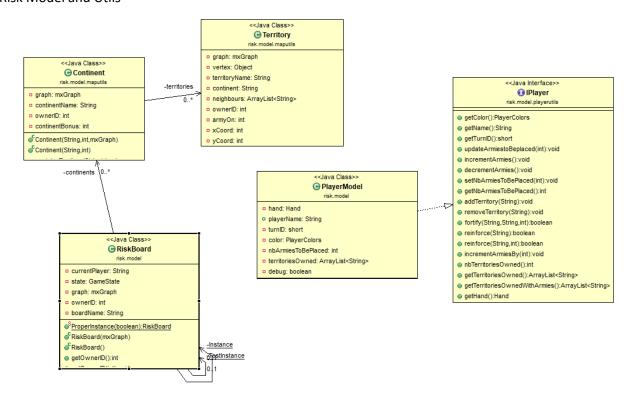
Risk Model

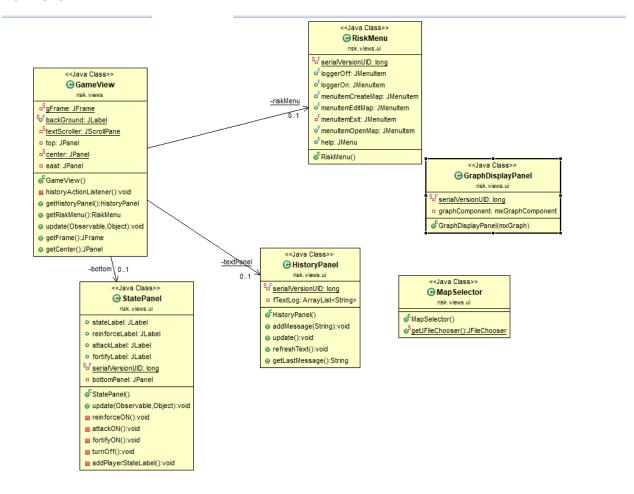




<<Java Class>> Territory risk.model.maputils graph: mxGraph <<Java Class>> vertex: Object Continent a territoryName: String risk.model.maputils continent: String graph: mxGraph neighbours: ArrayList<String> a continentName: String ownerID: int ownerID: int armyOn: int a continentBonus: int xCoord: int Continent(String,int,mxGraph) p yCoord: int Continent(String,int) Territory(String,String,String[],mxGraph,int,int) containsTerritory(String):boolean *Territory(String,String,mxGraph,int,int) getTerritory(String):Territory Territory(String,String,String[]) addTerritory(String,int,int):void Territory(String,String) addTerritory(String):void -territories getVertex():Object addTerritory(String,String[]):void getTerritoryName():String 0...* addTerritory(String,String[],int,int):void getContinentName():String getOwnerID():int getOwnerID():int setOwnerID(int):void setOwnerID(int):void getContinentName():String addNeighbours(String):void getContinentBonus():int getNeighbours():ArrayList<String> isOwned():boolean getArmyOn():int toString():String setArmyOn(int):void territoriesToString():String removeNeighbours(String):void getTerritories():ArrayList<String> toString():String clear():void territoriesToString():String removeTerritory(String):void getxCoord():int validateContinent():boolean setxCoord(int):void checkOwnerStatus():void getyCoord():int setyCoord(int):void clear():void validateTerritory():boolean

Risk Model and Utils





All With dependencies

